



Subject card

Subject name and code	Software engineering, PG_00045302						
Field of study	Data Engineering						
Date of commencement of studies	October 2024	Academic year of realisation of subject			2025/2026		
Education level	first-cycle studies	Subject group			Obligatory subject group in the field of study Subject group related to scientific research in the field of study		
Mode of study	Full-time studies	Mode of delivery			at the university		
Year of study	2	Language of instruction			English		
Semester of study	3	ECTS credits			3.0		
Learning profile	general academic profile	Assessment form			exam		
Conducting unit	Department of Software Engineering -> Faculty of Electronics, Telecommunications and Informatics						
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Aleksander Jarzębowicz					
	Teachers	dr inż. Aleksander Jarzębowicz					
Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	30.0	0.0	0.0	45
	E-learning hours included: 0.0						
Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan		Participation in consultation hours		Self-study	SUM
	Number of study hours	45		6.0		24.0	75
Subject objectives	The aim of the course is to introduce students to analysis and design as part of overall software project activities and to enable practical learning of UML as a tool for object-oriented analysis and design of IT systems.						
Learning outcomes	Course outcome		Subject outcome			Method of verification	
Subject contents	<ol style="list-style-type: none">1. Introduction2. Scope and subject of software engineering. Essential motivations and concepts.3. Areas of software engineering - an overview4. Requirements engineering: requirements elicitation, analysis and validation5. Requirements engineering: requirements specification6. Conceptual modelling. Languages for modelling and specification.7. Use cases8. Object-oriented analysis using UML9. Modelling of logical system structure: class diagrams10. Modelling of system structure: other structural diagrams11. Modelling system dynamics: sequence and communication diagrams12. Modelling system dynamics: representing object's state13. Design: system architecture14. Design: system (high-level) design and class (low-level) design15. Software reuse and design patterns16. Software development models (software lifecycle models)17. Software development methodologies (outline)						
Prerequisites and co-requisites							
Assessment methods and criteria	Subject passing criteria		Passing threshold		Percentage of the final grade		
	Lab		50.0%		50.0%		
	Exam		50.0%		50.0%		
Recommended reading	Basic literature		<ol style="list-style-type: none">1. Pressman R., Software Engineering: a Practitioner's Approach, 8th edition, McGraw-Hill, 20142. Booch G., Rumbaugh J., Jacobsen I.: The Unified Modeling Language User Guide (2nd Edition), Addison-Wesley, 2005				

	Supplementary literature	<ol style="list-style-type: none"> 1. Sommerville I., Software Engineering, 9th edition, Addison-Wesley, 2010 2. Maciaszek L.: Requirements analysis and system design, Addison-Wesley, 2007 3. Fowler M., Scott K.: UML distilled 3rd ed, Addison-Wesley, 2003 4. McLaughlin B., Pollice G., West D., Head First: Object-Oriented Analysis and Design, O'Reilly Media, 2006
	eResources addresses	Adresy na platformie eNauzanie:
Example issues/ example questions/ tasks being completed	<ul style="list-style-type: none"> • Draw a UML diagram (e.g. use case diagram, class diagram, state diagram) reflecting a given description of system requirements. • Describe a given software development model and discuss its strong and weak aspects. • Enumerate and describe requirements specification techniques. 	
Work placement	Not applicable	

Document generated electronically. Does not require a seal or signature.